

THE PROBABILISTIC STRUCTURE OF DISCRETE AGENT-BASED MODELS

ABSTRACT. This paper describes a formalization of agent-based models (ABMs) as random walks on regular graphs and relates the symmetry group of those graphs to a coarse-graining of the ABM that is still Markovian. An ABM in which N agents can be in δ different states leads to a Markov chain with δ^N states. In ABMs with a sequential update scheme by which one agent is chosen to update its state at a time, transitions are only allowed between system configurations that differ with respect to a single agent. This characterizes ABMs as random walks on regular graphs. The non-trivial automorphisms of those graphs make visible the dynamical symmetries that an ABM gives rise to because sets of micro configurations can be interchanged without changing the probability structure of the random walk. This allows for a systematic loss-less reduction of the state space of the model.

CONTENTS

1. Introduction	1
2. The Grammar of an ABM	3
3. From Functional Graphs to Markov Chains	5
4. Single-Step Dynamics and Random Walks on Regular Graphs	6
5. Graph Symmetries and Markov Chain Aggregation	8
6. Groups of Automorphisms, Macro Chains and System Properties	10
7. Summary	14
References	14

1. INTRODUCTION

Agent-based models (ABMs) are an attempt to understand how macroscopic regularities may emerge through processes of self-organization in systems of interacting agents. The approach is first and foremost a computational methodology and the mathematical formalization of ABMs is still in its infancy. This is probably due to the fact that a major motivation in the development of ABMs has been to relax a series of unrealistic assumptions made in other modeling frameworks just in order to keep mathematical tractability; namely, rationality, perfect information, agent homogeneity, among others. The other side of the coin is that the focus on computer models and algorithms makes difficult the comparison of different models and also complicates a rigorous analysis of the model behavior. In fact, the

problems of code verification and model comparison including the discussion of standards for the replication of ABMs have nowadays become an area of research in its own (see e.g., [12, 11, 9]). Many of those issues would probably be resolved with a sound mathematical formulation of an ABM. On the other hand, it is also clear that the precise mathematical specification of a high-dimensional system of heterogeneous interacting agents along with their update mechanisms can be cumbersome.

Agent-based systems are dynamical systems. Typically implemented on a computer, the time evolution is computed as an iterative process – an algorithm – in which agents are updated according to the specified rules. ABMs usually also involve a certain amount of stochasticity, because the agent choice and sometimes also the choice among different behavioral options is random. This is why Markov chain theory is a good candidate for the mathematical formalization of ABMs.

To the authors knowledge, the first systematic approach to the development of mathematical formalism for ABMs in general is due to Laubenbacher and co-workers. Ref. [16] reviews existing formal frameworks that have the potential to model ABMs, such as cellular automata and finite dynamical systems and argue for the latter as an appropriate mathematical framework. The possibility of using Markov chains in the analysis of ABMs has been pointed out in [13]. The main idea is to consider all possible configurations of the agent system as the state space Σ of a huge Markov chain. While Ref. [13] mainly relies on numerical computations to estimate the stochastic transition matrices of the models, we have shown how to derive explicitly the transition probabilities \hat{P} in terms of the update function \mathbf{u} and a probability distribution ω accounting for the stochastic parts in the model ([3, 4]). From general ABM to a particular class of models we refer to as single-step dynamics, this paper discusses in detail how to derive a microscopic Markov chain description (micro chain). It turns out that ABMs with a sequential update scheme can be conceived as random walks on regular graphs.

This, in turn, hints at the possibility of reducing the state space of the microscopic Markov chain by exploiting systematically the dynamical symmetries that an ABM gives rise to. Namely, the existence of non-trivial automorphisms of the micro chain tells us that certain sets of micro configurations can be interchanged without changing the probability structure of the random walk. These sets of micro states can be aggregated or lumped into a single macro state and the resulting macro-level process is still a Markov chain. In Markov chain theory, such a state space reduction by which no information about the dynamical behavior is lost is known as lumpability [6, 20, 14, 19, 5, 10].

Throughout the paper we use the Voter Model (VM from now on) as a simple paradigmatic example ([15, 7], among many others). In the VM, agents can adopt two different states, which we may denote as white \square and black \blacksquare . The attribute could account for the opinion of an agent regarding a certain issue, its approval or disapproval regarding certain attitudes. In an

economic context \blacksquare and \square could encode two different behavioral strategies, or, in a biological context, the occurrence of mutants in a population of individuals. The iteration process implemented by the VM is very simple. At each time step, an agent i is chosen at random along with one of its neighboring agents j and one of them imitates the state of the other (by convention we assume the first to imitate the second). In the long run, the model leads to a configuration in which all agents have adopted the same state (either \square or \blacksquare). In the context of biological evolution, this has been related to the fixation or extinction of a mutant in a population. The VM has also been interpreted as a simplistic form of a social influence process by which a shared convention is established in the entire population.

This paper is organized as follows. In Section 2, we discuss the general structure of ABMs in form of a graph of the possible model transitions. Markovianity of the ABM process on the graph can be established by a so-called random mapping representation (Section 3). In Section 4 a class of models giving rise to single-step dynamics and therefore to random walks on regular graphs is discussed. Section 5 relates the symmetries in those graphs to partitions of the micro process with respect to which the chain is lumpable and Section 6 illustrates this at the example of a single-step model with N agents that can be in three different states. We summarize these results in Section 7.

2. THE GRAMMAR OF AN ABM

Let us consider an abstract ABM with finite configuration space $\Sigma = \mathbf{S}^N$ (meaning that there are N agents with attributes $x_i \in \mathbf{S}$). Any iteration of the model (any run of the ABM algorithm) maps a configuration $\mathbf{x} \in \Sigma$ to another configuration $\mathbf{y} \in \Sigma$. In general, the case that no agent changes such that $\mathbf{x} = \mathbf{y}$ is also possible. Let us denote such a mapping by $F_z : \Sigma \rightarrow \Sigma$ and denote the set of all possible mappings by \mathcal{F} . Notice that any element of \mathcal{F} can be seen as a word of length $|\Sigma|$ over an $|\Sigma|$ -ary alphabet, and there are $|\Sigma|^{|\Sigma|}$ such words [8]:3.

Any $F_z \in \mathcal{F}$ induces a directed graph (Σ, F_z) the nodes of which are the elements in Σ (i.e., the agent configurations) and edges the set of ordered pairs $(\mathbf{x}, F_z(\mathbf{x}))$, $\forall \mathbf{x} \in \Sigma$. Such a graph is called functional graph of F_z because it displays the functional relations of the map F_z on Σ . That is, it represents the logical paths induced by F_z on the space of configurations for any initial configuration \mathbf{x} .

Each iteration of an ABM can be thought of as a stochastic choice out of a set of deterministic options. For an ABM in a certain configuration \mathbf{x} , there are usually several options (several \mathbf{y}) to which the algorithm may lead with a well-defined probability (see Fig. 1). Therefore, in an ABM, the transitions between the different configurations $\mathbf{x}, \mathbf{y}, \dots \in \Sigma$ are not defined by one single map F_z , but there is rather a subset $\mathcal{F}_Z \subset \mathcal{F}$ of maps out of which one map is chosen at each time step with a certain probability.

Let us assume we know all the mappings $\mathcal{F}_Z = \{F_1, \dots, F_z, \dots, F_n\}$ that are realized by the ABM of our interest. With this, we are able to define a functional graph representation by (Σ, \mathcal{F}_Z) which takes as the nodes all elements of Σ (all agent configurations) and an arc (\mathbf{x}, \mathbf{y}) exists if there is at least one $F_z \in \mathcal{F}_Z$ such that $F_z(\mathbf{x}) = \mathbf{y}$. This graph defines the »grammar« of the system for it displays all the logically possible transitions between any pair of configurations of the model.

Consider the VM with three agents as an example. In the VM agents have two possible states ($\mathbf{S} = \{\square, \blacksquare\}$) and the configuration space for a model of three agents is $\Sigma = \{\square, \blacksquare\}^3$. In the iteration process, one agent i is chosen at random along with one of its neighbors j and agent i imitates the state of j . This means that $y_i = x_j$ after the interaction event. Notice that once an agent pair (i, j) is chosen the update is defined by a deterministic map $\mathbf{u} : \mathbf{S}^2 \rightarrow \mathbf{S}$. Stochasticity enters first because of the random choice of i and second through the random choice of one agent in the neighborhood. Let us look at an example with three agents in the configuration $\mathbf{x} = (\square \blacksquare \blacksquare)$. If the first agent is chosen ($i = 1$ and $x_1 = \square$) then this agent will certainly change state to $y_1 = \blacksquare$ because it will in any case meet a black agent. For the second and the third agent ($i = 2$ or $i = 3$) the update result depends on whether one or the other neighbor is chosen because they are in different states. Noteworthy, different agent choices may lead to the same configuration. Here, this is the case if the agent pair $(2, 3)$ or $(3, 2)$ is chosen in which case the agent $(2 \text{ or } 3)$ does not change its state because $x_2 = x_3$. Therefore we have $\mathbf{y} = \mathbf{x}$ and there are two paths realizing that transition.

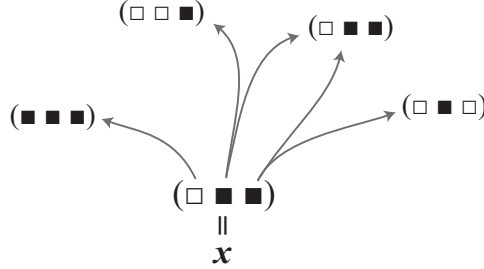


FIGURE 1. Possible paths from configuration $\mathbf{x} = (\square \blacksquare \blacksquare)$ in a small VM of three agents.

In practice, the explicit construction of the entire functional graph may rapidly become a tedious task due to the huge dimension of the configuration space and the fact that one needs to check if $F_z(\mathbf{x}) = \mathbf{y}$ for each mapping $F_z \in \mathcal{F}_Z$ and all pairs of configurations \mathbf{x}, \mathbf{y} . On the other hand, the main interest here is a theoretical one, because, as a matter of fact, a representation as a functional graph of the form $\Gamma = (\Sigma, \mathcal{F}_Z)$ exists for any model that comes in form of an iterated computer algorithm. It is therefore a quite general way of formalizing ABMs and, as we will see in the sequel, it allows under some conditions to verify the Markovianity of the models at the micro level.

3. FROM FUNCTIONAL GRAPHS TO MARKOV CHAINS

A functional graph $\Gamma = (\Sigma, \mathcal{F}_Z)$ defines the »grammar« of an ABM in the sense that it shows all possible transitions enabled by the model. It is the first essential step in the construction of the Markov chain associated with the ABM at the micro level because there is a non-zero transition probability only if there is an arrow in the functional graph. Consequently, all that is missing for a Markov chain description is the computation of the respective transition probabilities.

For a class of models, including the VM, this is relatively simple because we can derive a random mapping representation [17]:6/7 directly from the ABM rules. Namely, if F_{z_1}, F_{z_2}, \dots is a sequence of independent random maps, each having the same distribution ω , and $S_0 \in \Sigma$ has distribution μ_0 , then the sequence S_0, S_1, \dots defined by

$$(1) \quad S_t = F_{z_t}(S_{t-1}), t \geq 1$$

is a Markov chain on Σ with transition matrix \hat{P} :

$$(2) \quad \hat{P}(\mathbf{x}, \mathbf{y}) = \Pr_{\omega}[z, F_z(\mathbf{x}) = \mathbf{y}; \mathbf{x}, \mathbf{y} \in \Sigma].$$

Conversely [17], any Markov chain has a random map representation (RMR). Therefore, in that case, (1) and (2) may be taken as an equivalent definition of a Markov chain. This is particularly useful in our case, because it shows that an ABM which can be described as above is, from a mathematical point of view, a Markov chain. This includes several models described in [13].

For general ABMs the explicit construction of a RMR can be cumbersome because it requires the dissection of stochastic and deterministic elements of the iteration procedure of the model. In the VM, this separation is clear-cut and therefore a RMR is obtained easily. In the VM, the random part consists of the choice of two connected agents (i, j) . Once this choice is made we know that $y_i = x_j$ by the interaction rule. This is sufficient to derive the »grammar« of the VM, because we need only to check one by one for all possible choices (i, j) which transitions this choice induces on the configuration space. For a system of three agents, with all agents connected to the other two, the set of functions $\mathcal{F}_Z = \{F_1, \dots, F_z, \dots, F_n\}$ is specified in Table 1. Notice that with three agents, there are 8 possible configurations indexed here by a, b, \dots, h . Moreover, there are 6 possible choices for (i, j) such that \mathcal{F}_Z consists of $n = 6$ mappings.

Each row of the table represents a mapping $F_z : \Sigma \rightarrow \Sigma$ by listing to which configuration \mathbf{y} the respective map takes the configurations a to h . The first row, to make an example, represents the choice of the agent pair $(1, 2)$. The changes this choice induces depend on the actual agent configuration \mathbf{x} . Namely, for any \mathbf{x} with $x_1 = x_2$ we have $F_1(\mathbf{x}) = F_{(1,2)}(\mathbf{x}) = \mathbf{x}$. So the configurations a, b, g, h are not changed by $F_{(1,2)}$. For the other configurations it is easy to see that $(\blacksquare\blacksquare\blacksquare) \rightarrow (\square\square\blacksquare)$ ($c \rightarrow g$), $(\square\blacksquare\blacksquare) \rightarrow (\blacksquare\blacksquare\blacksquare)$ ($d \rightarrow a$), $(\blacksquare\square\square) \rightarrow (\square\square\square)$ ($e \rightarrow h$), and $(\square\blacksquare\square) \rightarrow (\blacksquare\blacksquare\square)$ ($f \rightarrow b$). Notice that the two configurations $(\square\square\square)$ and $(\blacksquare\blacksquare\blacksquare)$ with all agents equal are not

z	(i, j)	a	b	c	d	e	f	g	h
		■ ■ ■	■ ■ □	■ □ ■	□ ■ ■	■ □ □	□ ■ □	□ □ ■	□ □ □
1	(1, 2)	a	b	g	a	h	b	g	h
2	(1, 3)	a	f	c	a	h	f	c	h
3	(2, 1)	a	b	a	g	b	h	g	h
4	(3, 1)	a	a	c	f	c	f	h	h
5	(2, 3)	a	e	a	d	e	h	d	h
6	(3, 2)	a	a	e	d	e	d	h	h

TABLE 1. \mathcal{F}_Z for the VM with three agents.

changed by any map and correspond therefore to the final configurations of the VM.

In the RMR, we can use the possible agent choices (i, j) in Table 1 directly to index the collection of maps $F_{(i,j)} \in \mathcal{F}_Z$. We denote as $\omega(i, j)$ the probability of choosing the agent pair (i, j) which corresponds to choosing the map $F_{(i,j)}$. It is clear that we can proceed in this way in all models where the stochastic part concerns only the choice of agents. Then, the distribution ω is independent of the current system configuration and the same for all times ($\omega(z_t) = \omega(z)$). In this case, we obtain for the transition probabilities

$$(3) \quad \hat{P}(\mathbf{x}, \mathbf{y}) = \Pr_{\omega}[(i, j), F_{(i,j)}(\mathbf{x}) = \mathbf{y}] = \sum_{\substack{(i,j): \\ F_{(i,j)}(\mathbf{x}) = \mathbf{y}}} \omega(i, j).$$

That is, the probability of transition from \mathbf{x} to \mathbf{y} is the conjoint probability $\sum \omega(i, j)$ of choosing an agent pair (i, j) such that the corresponding map takes \mathbf{x} to \mathbf{y} (i.e., $F_{(i,j)}(\mathbf{x}) = \mathbf{y}$).

4. SINGLE-STEP DYNAMICS AND RANDOM WALKS ON REGULAR GRAPHS

In the sequel, we focus on a class of models which we refer to as *single-step dynamics*. They are characterized by the fact that only one agent changes at a time step. Notice that this is very often the case in ABMs with a sequential update scheme and that sequential update is, as a matter of fact, the most typical iteration scheme in ABMs. In terms of the »grammar« of these models, this means that non-zero transition probabilities are only possible between system configuration that differ in at most one position. And this gives rise to random walks on regular graphs.

Consider a set of N agents each one characterized by individual attributes x_i that are taken in a finite list of possibilities $\mathbf{S} = \{1, \dots, \delta\}$. In this case, the space of possible agent configurations is $\Sigma = \mathbf{S}^N$. Consider further a deterministic update function $\mathbf{u} : \mathbf{S}^r \times \Lambda \rightarrow \mathbf{S}$ which takes configuration $\mathbf{x} \in \Sigma$ at time t to configuration $\mathbf{y} \in \Sigma$ at $t + 1$ by

$$(4) \quad y_i = \mathbf{u}(x_i, x_j, \dots, x_k, \lambda).$$

To go from one time step to the other in agent systems, usually, an agent i is chosen first to perform a step. The decision of i then depends on its current state (x_i) and the attributes of its neighbors (x_j, \dots, x_k). The finite set Λ accounts for a possible stochastic part in the update mechanism such that different behavioral options are implemented by different update functions $\mathbf{u}(\dots, \lambda_1)$, $\mathbf{u}(\dots, \lambda_2)$ etc. Notice that for the case in which the attributes of the agents (x_i, x_j, \dots, x_k) uniquely determine the agent decision we have $\mathbf{u} : \mathbf{S}^r \rightarrow \mathbf{S}$ which strongly resembles the update rules implemented in cellular automata (CA).

As opposed to classical CA, however, a sequential update scheme is used in the class of models considered here. In the iteration process, first, a random choice of agents along with a λ to index the possible behavioral options is performed with probability $\omega(i, j, \dots, k, \lambda)$. This is followed by the application of the update function which leads to the new state of agent i by Eq. (4).

Due to the sequential application of an update rule of the form $\mathbf{u} : \mathbf{S}^r \times \Lambda \rightarrow \mathbf{S}$ only one agent (namely agent i) changes at a time so that all elements in \mathbf{x} and \mathbf{y} are equal except that element which corresponds to the agent that was updated during the step from \mathbf{x} to \mathbf{y} . Therefore, $x_j = y_j, \forall j \neq i$ and $x_i \neq y_i$. We call \mathbf{x} and \mathbf{y} adjacent and denote this by $\mathbf{x} \stackrel{i}{\sim} \mathbf{y}$.

It is then also clear that a transition from \mathbf{x} to \mathbf{y} is possible if $\mathbf{x} \sim \mathbf{y}$. Therefore, the adjacency relation \sim defines the »grammar« Γ_{SSD} of the entire class of single-step models. Namely, the existence of a map F_z that takes \mathbf{x} to \mathbf{y} , $\mathbf{y} = F_z(\mathbf{x})$, implies that $\mathbf{x} \stackrel{i}{\sim} \mathbf{y}$ for some $i \in \{1, \dots, N\}$. This means that any ABM that belongs to the class of single-step models performs a walk on Γ_{SSD} or on a subgraph of it.

Let us briefly consider the structure of the graph Γ_{SSD} associated to the entire class of single-step models. From $\mathbf{x} \stackrel{i}{\sim} \mathbf{y}$ for $i = 1, \dots, N$ we know that for any \mathbf{x} , there are $(\delta - 1)N$ different vectors \mathbf{y} which differ from \mathbf{x} in a single position, where δ is the number of possible agent attributes. Therefore, Γ_{SSD} is a regular graph with degree $(\delta - 1)N + 1$, because in our case, the system may loop by $y_i = x_i$. As a matter of fact, our definition of adjacency as »different in one position of the configuration« is precisely the definition of so-called Hamming graphs which tells us that $\Gamma_{SSD} = H(N, \delta)$ (with loops). In the case of the VM, where $\delta = 2$ we find $H(N, 2)$ which corresponds to the N -dimensional hypercube.

As before, the transition probability matrix of the micro chain is denoted by \hat{P} with $\hat{P}(\mathbf{x}, \mathbf{y})$ being the probability for the transition from \mathbf{x} to \mathbf{y} . The previous considerations tell us that non-zero transition probabilities can exist only between two configurations that are linked in $H(N, \delta)$ plus the loop ($\hat{P}(\mathbf{x}, \mathbf{x})$). Therefore, each row of \hat{P} contains no more than $(\delta - 1)N + 1$ non-zero entries. In the computation of \hat{P} we concentrate on pairs of adjacent

configurations. For $\mathbf{x} \stackrel{i}{\sim} \mathbf{y}$ with $x_i \neq y_i$ we have

$$(5) \quad \hat{P}(\mathbf{x}, \mathbf{y}) = \sum_{\substack{(i,j,\dots,k,\lambda): \\ y_i = \mathbf{u}(x_i, x_j, \dots, x_k, \lambda)}} \omega(i, j, \dots, k, \lambda)$$

which is the conjoint probability to choose agents and a rule $(i, j, \dots, k, \lambda)$ such that the i th agent changes its attribute by $y_i = \mathbf{u}(x_i, x_j, \dots, x_k, \lambda)$. For the probability that the model remains in \mathbf{x} , $\hat{P}(\mathbf{x}, \mathbf{x})$, we have

$$(6) \quad \hat{P}(\mathbf{x}, \mathbf{x}) = 1 - \sum_{\mathbf{y} \sim \mathbf{x}} \hat{P}(\mathbf{x}, \mathbf{y}).$$

Eq. (5) makes visible that the probability distribution ω plays the crucial role in the computation of the elements of \hat{P} , a fact that has been exploited in Ref. [2].

5. GRAPH SYMMETRIES AND MARKOV CHAIN AGGREGATION

Markov chain aggregation concerns the question of what happens when the micro-level process – defined by the micro chain (Σ, \hat{P}) – is projected onto a coarser partition of the state space Σ . Such a situation naturally arises if the ABM is observed not at the micro level of Σ , but rather in terms of a measure ϕ on Σ by which all configuration in Σ that give rise to the same measurement are mapped into the same macro state, say $X_k \in \mathbf{X}$. The first important question then concerns the *lumpability* of the micro chain with respect to the partition \mathbf{X} . In the case of lumpability, the resulting macro-level process is still a Markov chain and the transition rates P can be obtained in a relatively simple way from the microscopic transition matrix \hat{P} . Fig. 2 illustrates such a projection construction.

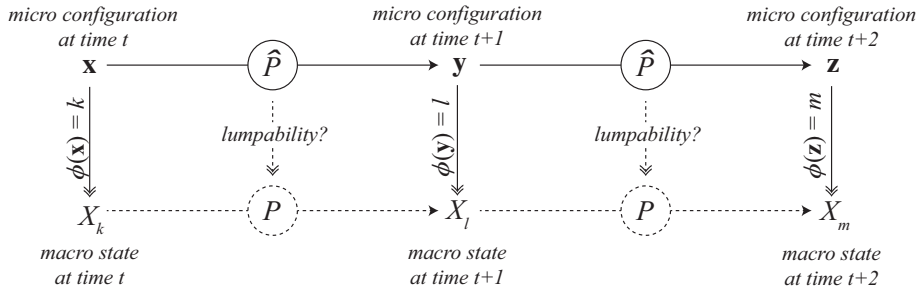


FIGURE 2. A micro process $(\mathbf{x}, \mathbf{y}, \mathbf{z} \in \Sigma)$ is observed (ϕ) at a higher level and this observation defines another macro level process $(X_k, X_l, X_m \in \mathbf{X})$. The micro process is a Markov chain with transition matrix \hat{P} . The macro process is a Markov chain (with P) only in the case of lumpability.

Necessary and sufficient conditions for lumpability are provided by Thm. 6.3.2 in [14]. Let us denote by $\hat{p}_{\mathbf{x}\mathbf{y}}$ the conjoint probability for \mathbf{x} to go to

elements $\mathbf{y} \in Y$ where $Y \subseteq \Sigma$ is a subset of the configuration space. Thm. 6.3.2 in [14] states that a Markov chain (\hat{P}, Σ) is *lumpable* with respect to a partition $\mathbf{X} = (X_1, \dots, X_P)$ if for every two subsets X_k and X_l the sum $\hat{p}_{\mathbf{x}X_l} = \sum_{\mathbf{y} \in X_l} \hat{p}_{\mathbf{x}\mathbf{y}}$ is equal for all $\mathbf{x} \in X_k$. Moreover, these common values form the transition probabilities $P(X_k, X_l)$ for a new chain (P, \mathbf{X}) .

Since ABM micro chains can be seen as random walks on regular graphs, it is convenient to provide lumpability conditions in form of the symmetry structure of the micro chain. Let us restate the respective result (previously introduced in a similar form in [3], Prop. 3.1):

Theorem 5.1. *Let (Σ, \hat{P}) be a Markov chain and \mathbf{x}, \mathbf{y} elements of Σ . Consider a partition \mathbf{X} of Σ and along with \mathbf{X} a transformation group \mathcal{G} acting on Σ that generates \mathbf{X} . (That is, the orbits of \mathcal{G} on Σ construct \mathbf{X} .) If the Markov transition probability \hat{P} is symmetric with respect to \mathcal{G} ,*

$$(7) \quad \hat{P}(\mathbf{x}, \mathbf{y}) = \hat{P}(\hat{\sigma}(\mathbf{x}), \hat{\sigma}(\mathbf{y})) : \forall \hat{\sigma} \in \mathcal{G},$$

the partition $\mathbf{X} = (X_1, \dots, X_n)$ is (strongly) lumpable.

Proof. For the proof it is sufficient to show that any two configurations \mathbf{x} and \mathbf{x}' with $\mathbf{x}' = \hat{\sigma}(\mathbf{x})$ satisfy

$$(8) \quad \hat{p}_{\mathbf{x}Y} = \sum_{\mathbf{y} \in Y} \hat{P}(\mathbf{x}, \mathbf{y}) = \sum_{\mathbf{y} \in Y} \hat{P}(\mathbf{x}', \mathbf{y}) = \hat{p}_{\mathbf{x}'Y}$$

for all $Y \in \mathbf{X}$. Consider any two subsets $X, Y \in \mathbf{X}$ and take $\mathbf{x} \in X$. Because \mathcal{G} preserves the partition it is true that $\mathbf{x}' \in X$. Now we have to show that Eq. (8) holds. First the probability for $\mathbf{x}' = \hat{\sigma}(\mathbf{x})$ to go to an element $\mathbf{y} \in Y$ is

$$(9) \quad \hat{p}_{\hat{\sigma}(\mathbf{x})Y} = \sum_{\mathbf{y} \in Y} \hat{P}(\hat{\sigma}(\mathbf{x}), \mathbf{y}).$$

Because the $\hat{\sigma}$ are bijections and preserve \mathbf{X} we have $\hat{\sigma}(Y) = Y$ and there is for every $\mathbf{y} \in Y$ exactly one $\hat{\sigma}(\mathbf{y}) \in Y$. Therefore we can substitute

$$(10) \quad \hat{p}_{\hat{\sigma}(\mathbf{x})Y} = \sum_{\mathbf{y} \in Y} \hat{P}(\hat{\sigma}(\mathbf{x}), \hat{\sigma}(\mathbf{y})) = \sum_{\mathbf{y} \in Y} \hat{P}(\mathbf{x}, \mathbf{y}) = \hat{p}_{\mathbf{x}Y},$$

where the second equality comes by the symmetry condition (7) that $\hat{P}(\mathbf{x}, \mathbf{y}) = \hat{P}(\hat{\sigma}(\mathbf{x}), \hat{\sigma}(\mathbf{y}))$. \square

The usefulness of the lumpability conditions stated in Thm. 5.1 becomes apparent recalling that ABMs can be seen as random walks on regular graphs defined by the functional graph or »grammar« of the model $\Gamma = (\Sigma, \mathcal{F}_Z)$. The full specification of the micro process (Σ, \hat{P}) is obtained by assigning transition probabilities to the connections in Γ and we can interpret this as a

weighted graph. The regularities of (Σ, \hat{P}) are captured by a number of non-trivial automorphisms which, in the case of ABMs, reflect the symmetries of the models.

In fact, Thm. 5.1 allows to systematically exploit the symmetries of an agent model in the construction of partitions with respect to which the micro chain is lumpable. Namely, the symmetry requirement in Thm. 5.1, that is, Eq. (7), corresponds precisely to the usual definition of automorphisms of (Σ, \hat{P}) . The set of all permutations $\hat{\sigma}$ that satisfy (7) corresponds then to the automorphism group of (Σ, \hat{P}) .

Lemma 5.2. *Let \mathcal{G} be the automorphism group of the micro chain (Σ, \hat{P}) . The orbits of \mathcal{G} define a lumpable partition \mathbf{X} such that every pair of micro configurations $\mathbf{x}, \mathbf{x}' \in \Sigma$ for which $\exists \hat{\sigma} \in \mathcal{G}$ such that $\mathbf{x}' = \hat{\sigma}(\mathbf{x})$ belong to the same subset $X_i \in \mathbf{X}$.*

Remark 5.1. *Lemma 5.2 actually applies to any \mathcal{G} that is a proper subgroup of the automorphism group of (Σ, \hat{P}) . The basic requirement for such a subset \mathcal{G} to be a group is that be closed under the group operation which establishes that $\hat{\sigma}(X_i) = X_i$. With the closure property, it is easy that any such subgroup \mathcal{G} defines a lumpable partition in the sense of Thm. 5.1.*

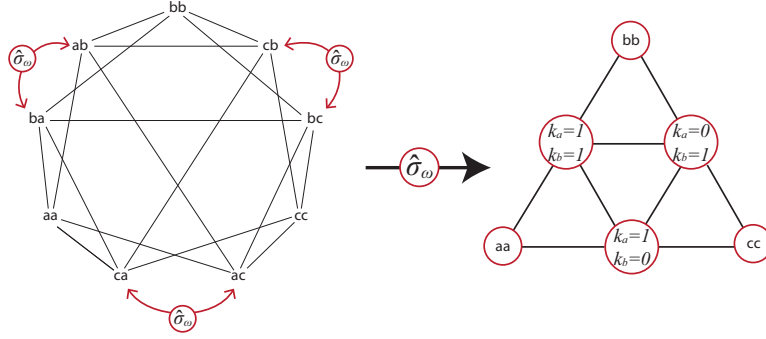
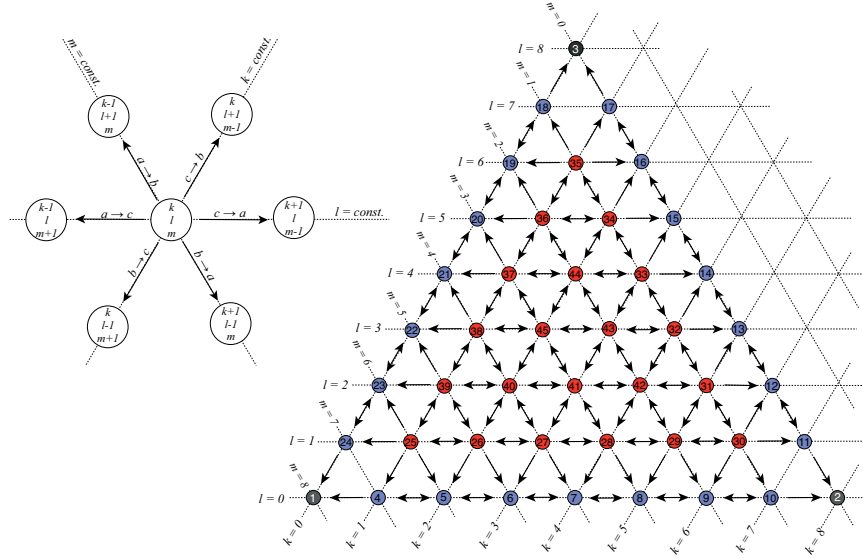
6. GROUPS OF AUTOMORPHISMS, MACRO CHAINS AND SYSTEM PROPERTIES

In this section we illustrate the previous ideas at the example of three state single-step dynamics. Consider a system of N agents each one characterized by an attribute $x_i \in \{a, b, c\}$, that is $\delta = 3$. As discussed in Section 4, the corresponding graph Γ encoding all the possible transitions is the Hamming graph $H(N, 3)$. The nodes \mathbf{x}, \mathbf{y} in $H(N, 3)$ correspond to all possible agent combinations and are written as vectors $\mathbf{x} = (x_1, \dots, x_N)$ with symbols $x_i \in \{a, b, c\}$. The automorphism group of $H(N, 3)$ is composed of two groups generated by operations changing the order of elements in the vector (agent permutations) and by permutations acting on the set of symbols $\mathbf{S} = \{a, b, c\}$ (agent attributes). Namely, it is given by the direct product

$$(11) \quad \text{Aut}(H(N, \delta)) = \mathcal{S}_N \otimes \mathcal{S}_\delta$$

of the symmetric group \mathcal{S}_N acting on the agents and the group \mathcal{S}_δ acting on the agent attributes.

Let us first look at a very small system of $N = 2$ agents and $\delta = 3$ states. The corresponding microscopic structure – the graph $H(2, 3)$ – is shown on the l.h.s. of Fig. 3. It also illustrates the action of \mathcal{S}_N on the $\mathbf{x}, \mathbf{y} \in \Sigma$, that is, the bijection induced on the configuration space by permuting the agent labels. Noteworthy, in the case of $N = 2$ there is only one alternative ordering of agents denoted here as $\hat{\sigma}_\omega(\mathbf{x})$ which takes $(x_1, x_2) \xleftrightarrow{\hat{\sigma}_\omega} (x_2, x_1)$. The respective group $\mathcal{S}_{N=2}$ therefore induces a partition in which all configurations \mathbf{x}, \mathbf{y} with the same number of attributes a, b, c are *lumped* into the same set, which we may denote as $X_{\langle k_a, k_b, k_c \rangle}$. See r.h.s. of Fig. 3.

FIGURE 3. $H(2,3)$ and the reduction induced by \mathcal{S}_N .FIGURE 4. Transition structure (l.h.s) and state topology (r.h.s) of the VM with three attributes for $N = 8$.

More generally in the case of N agents and δ agent attributes the group \mathcal{S}_N induces a partition of the configuration space Σ by which all configurations with the same attribute frequencies are collected in the same macro set. Let us define $N_s(\mathbf{x})$ to be the number of agents in the configuration \mathbf{x} with attribute s , $s = 1, \dots, \delta$, and then $X_{\langle k_1, k_2, \dots, k_\delta \rangle} \subset \Sigma$ as

$$(12) \quad X_{\langle k_1, \dots, k_s, \dots, k_\delta \rangle} = \left\{ \mathbf{x} \in \Sigma : N_1(\mathbf{x}) = k_1, \dots, N_s(\mathbf{x}) = k_s, \dots \right. \\ \left. \dots, N_\delta(\mathbf{x}) = k_\delta \text{ and } \sum_{s=1}^{\delta} k_s = N \right\}.$$

Each $X_{\langle k_1, k_2, \dots, k_\delta \rangle}$ contains all the configurations \mathbf{x} in which exactly k_s agents hold attribute s for any s . We use the notation $\langle k_1, k_2, \dots, k_\delta \rangle$ to indicate that $\sum_{s=1}^{\delta} k_s = N$. Therefore, the reduced state space is organized as a δ -simplex lattice, see Fig. 4.

For a model with $N = 8$ and $\delta = 3$ the resulting reduced state space is shown in Fig. 4. The transition structure depicted in Fig. 4 corresponds to the VM. The number of a , b and c agents is denoted by (respectively) k , l and m so that $\mathbf{X} = \{X_{\langle k, l, m \rangle} : 0 \leq k, l, m \leq N, k + l + m = N\}$. The number of states for a system with N agents is $S = \sum_{i=0}^N (i + 1) = \frac{(N+1)(N+2)}{2}$.

For Voter-like models – used, for instance, as models of opinion and social dynamics – it is not unusual to study the dynamical behavior by looking at the time evolution of the respective attribute frequencies. It is important to notice, however, that the resulting partition is lumpable only if the transition matrix \hat{P} is symmetric with respect to the action of \mathcal{S}_N on Σ , namely if Thm. 5.1 holds for \mathcal{S}_N . We have shown in [3] that this is only true for homogeneous mixing and the case of inhomogeneous interaction topologies is discussed in [2].

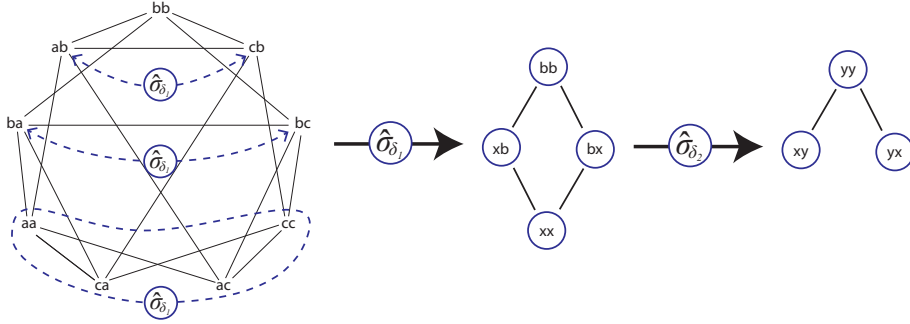


FIGURE 5. $H(2,3)$ and the reductions induced by \mathcal{S}_δ .

Let us now consider \mathcal{S}_δ . On the l.h.s. of Fig. 5 the graph $H(2,3)$ is shown along with the bijections on it induced by permutation of attributes a and c , $abc \xleftrightarrow{\hat{\sigma}_{\delta_1}} cba$). Effectively, this corresponds to the situation of looking at »one attribute (b) against the other two ($x = a \cup c$)«. Noteworthy, taking that perspective (see graph in the middle of Fig. 5) corresponds to a reduction of $H(2,3)$ to $H(2,2)$ or, more generally, of $H(N,3)$ to the hypercube $H(N,2)$. This means that, under the assumption of symmetric agent rules with respect to the attributes, single-step models with δ states are reducible to the binary case.

Moreover, even the binary case allows for further reduction (see r.h.s. of Fig. 5). Namely, assuming the additional symmetry $bx \xleftrightarrow{\hat{\sigma}_{\delta_2}} xb$ corresponding in a binary setting to the simultaneous flip of all agent states $x_i \rightarrow \bar{x}_i, \forall i$. The VM is a nice example in which independent of the interaction topology,

$\hat{P}(\mathbf{x}, \mathbf{y}) = \hat{P}(\bar{\mathbf{x}}, \bar{\mathbf{y}})$. This reduces the state space to one half of $H(N, 2)$, which we shall denote as $H_{1/2}(N, 2)$.

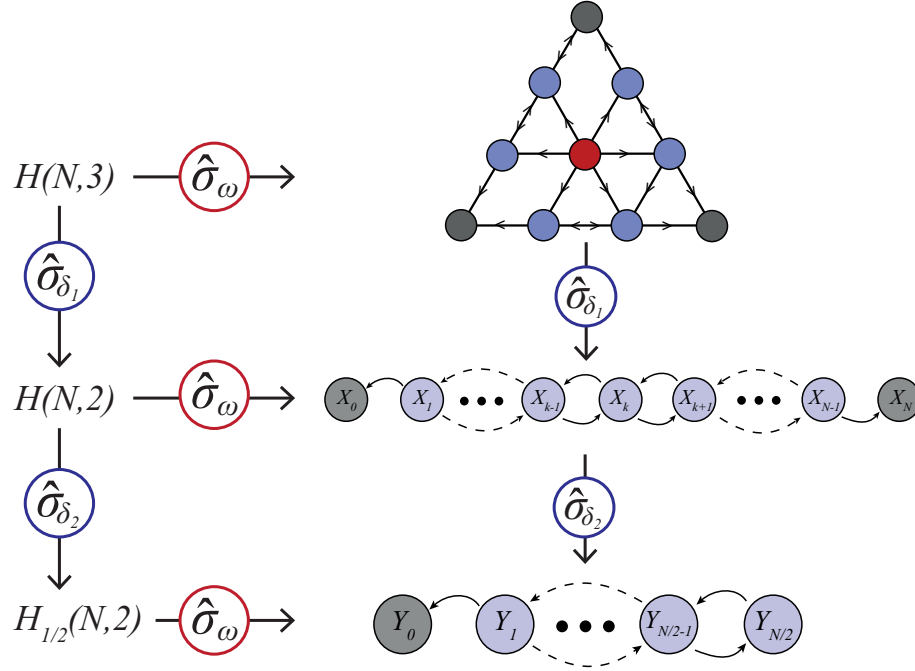


FIGURE 6. Different levels of description are associated to different symmetry groups of $H(N, 3)$.

The most interesting reductions can be reached by the combination of \mathcal{S}_N and \mathcal{S}_δ . Fig. 6 shows possible combinations and the resulting macroscopic state spaces starting from $H(N, 3)$. For instance, partitioning $H(N, 3)$ by using the set of agent permutations \mathcal{S}_N leads to state space organized as a triangular lattice (see also Fig. 4). Lumpability of the micro process (Σ, \hat{P}) on $H(N, 3)$ with respect to this state space rests upon the symmetry of the agent interaction probabilities with respect to all agent permutations ([2], see [1] for a discussion of the non-lumpable case). From the triangular structure shown on the upper right in Fig. 6, a further reduction can be obtained by taking into account the symmetry of the interaction rules with respect to (at least) one pair of attributes, which we have denoted as $\hat{\sigma}_{\delta_1}$. The resulting macro process on $\mathbf{X} = (X_0, \dots, X_N)$ is a random walk on the line with $N + 1$ states, known as Moran process for the VM interaction (after [18]). In a binary setting, the macro states X_k collect all micro configurations with k agents in state \square (and therefore $N - k$ agents in \blacksquare). Notice that a Markov projection to the Moran process is possible also for $\delta > 3$ if the micro process is symmetric with respect to permutations of (at least) $\delta - 1$ attributes. The group of transformations associated to this partition may be written as $\mathcal{S}_N \otimes \mathcal{S}_{\delta-1} \subset \text{Aut}(H(N, \delta))$.

The reduction obtained by using the full automorphism group of $H(N, 3)$ is shown on the bottom of Fig. 6. With respect to the Moran process on $\mathbf{X} = (X_0, \dots, X_N)$, it means that the pairs $\{X_k, X_{(N-k)}\}$ are lumped into the same state Y_k . This can be done if we have for any k , $P(X_k, X_{k\pm 1}) = P(X_{(N-k)}, X_{(N-k)\mp 1})$. As a matter of fact, this description still captures the number of agents in the same state, but now information about in which state they are is omitted. This is only possible (lumpable) if the model implements completely symmetric interaction rules.

7. SUMMARY

This paper analyses the probabilistic structure of a class of agent-based models (ABMs). In an ABM in which N agents can be in δ different states there are δ^N possible agent configurations and each iteration of the model takes one configuration into another one. It is therefore convenient to conceive of the agent configurations as the nodes of a huge directed graph and to link two configurations \mathbf{x}, \mathbf{y} whenever the application of the ABM rules to \mathbf{x} may lead to \mathbf{y} in one step. If a model operates with a sequential update scheme by which one agent is chosen to update its state at a time, transitions are only allowed between system configurations that differ with respect to a single element (agent). The graph associated to those single-step models is the Hamming graph $H(N, \delta)$.

The fact that a single-step ABM corresponds to a random walk on a regular graph allows for a systematic study of the symmetries in the dynamical structure of an ABM. Namely, the existence of non-trivial automorphisms of the ABM micro chain tells us that certain sets of agent configurations can be interchanged without changing the probability structure of the random walk. These sets of micro states can be aggregated or lumped into a single macro state and the resulting macro-level process is still a Markov chain. If the microscopic rules are symmetric with respect agent (\mathcal{S}_N) and attribute (\mathcal{S}_δ) permutations the full automorphism group of $H(N, \delta)$ is realized and allows for a reduction from δ^N micro to around $N/2$ macro states. Moreover, different combinations of subgroups of automorphisms and the reductions they imply are rather meaningful in terms of observables and system properties.

Notice finally that other update schemes (beyond single-step dynamics) – even the case of synchronous update¹ – do not necessarily affect the symmetries of the micro process. The described approach may be applied to these cases as well. Extending the framework to models with continuous agent attributes is another challenging issue to be addressed by future work.

REFERENCES

- [1] S. Banisch. From Microscopic Heterogeneity to Macroscopic Complexity in the Contrarian Voter Model, 2014. Submitted.

¹The author thanks Jürgen Jost for this suggestion.

- [2] S. Banisch and R. Lima. Markov Chain Aggregation for Simple Agent-Based Models on Symmetric Networks: The Voter Model, 2013. Submitted.
- [3] S. Banisch, R. Lima, and T. Araújo. Agent Based Models and Opinion Dynamics as Markov Chains. *Social Networks*, 34:549–561, 2012.
- [4] S. Banisch, R. Lima, and T. Araújo. Aggregation and Emergence in Agent-Based Models: A Markov Chain Approach. In T. Gilbert, M. Kirkilionis, and G. Nicolis, editors, *Proceedings of the European Conference on Complex Systems 2012*, Springer Proceedings in Complexity, pages 3–7. Springer International Publishing, 2013.
- [5] P. Buchholz. Exact and Ordinary Lumpability in Finite Markov Chains. *J. Appl. Prob.*, 31(1):59–75, 1994.
- [6] C. J. Burke and M. Rosenblatt. A Markovian Function of a Markov Chain. *The Annals of Mathematical Statistics*, 29(4):1112 – 1122, 1958.
- [7] C. Castellano, S. Fortunato, and V. Loreto. Statistical physics of social dynamics. *Reviews of Modern Physics*, 81(2):591–646, 2009.
- [8] P. Flajolet and A. M. Odlyzko. Random Mapping Statistics. In *Advances in Cryptology*, pages 329–354. Springer Verlag, 1990.
- [9] J. M. Galán, L. R. Izquierdo, S. S. Izquierdo, J. I. Santos, R. del Olmo, A. López-Paredes, and B. Edmonds. Errors and Artefacts in Agent-Based Modelling. *Journal of Artificial Societies and Social Simulation*, 12(1):1, 2009.
- [10] O. Görnerup and M. N. Jacobi. A Method for Inferring Hierarchical Dynamics in Stochastic Processes. *Advances in Complex Systems*, 11(1):1–16, 2008.
- [11] V. Grimm, U. Berger, F. Bastiansen, S. Eliassen, V. Ginot, J. Giske, J. Goss-Custard, T. Grand, S. K. Heinz, G. Huse, A. Huth, J. U. Jepsen, C. Jorgensen, W. M. Mooij, B. Muller, G. Pe’er, C. Pious, S. F. Railsback, A. M. Robbins, M. M. Robbins, E. Rossmanith, N. Ruger, E. Strand, S. Souissi, R. A. Stillman, R. Vabo, U. Visser, and D. L. Deangelis. A standard protocol for describing individual-based and agent-based models. *Ecological Modelling*, 198:115–126, 2006.
- [12] D. Hales, J. Rouchier, and B. Edmonds. Model-to-Model Analysis. *Journal of Artificial Societies and Social Simulation*, 6(4), 2003.
- [13] L. R. Izquierdo, S. S. Izquierdo, J. M. Galán, and J. I. Santos. Techniques to Understand Computer Simulations: Markov Chain Analysis. *Journal of Artificial Societies and Social Simulation*, 12(1):6, 2009.
- [14] J. G. Kemeny and J. L. Snell. *Finite Markov Chains*. Springer, 1976.
- [15] M. Kimura and G. H. Weiss. The stepping stone model of population structure and the decrease of genetic correlation with distance. *Genetics*, 49:561–576, 1964.
- [16] R. C. Laubenbacher, A. S. Jarrah, H. S. Mortveit, and S. S. Ravi. Agent Based Modeling, Mathematical Formalism for. In *Encyclopedia of Complexity and Systems Science*, pages 160–176. 2009.
- [17] D. A. Levin, Y. Peres, and E. L. Wilmer. *Markov chains and mixing times*. American Mathematical Society, 2009.
- [18] P. A. P. Moran. Random processes in genetics. In *Proceedings of the Cambridge Philosophical Society*, volume 54, pages 60–71, 1958.
- [19] L. C. G. Rogers and J. W. Pitman. Markov Functions. *The Annals of Probability*, 9(4):573–582, 1981.
- [20] M. Rosenblatt. Functions of a Markov Process that are Markovian. *Journal of Mathematics and Mechanics*, 8(4):134 –145, 1959.